<u>In the Claims:</u>

1.   (Cancelled)

2.   (Currently Amended) ~~The system of claim 1 wherein said preprocessor performs the steps of:~~ A system to migrate a software application from one application server to a different application server, comprising:

        a first application server that includes an application deployed thereon;

        a preprocessor server component on the first application server used to interrogate the functionality of the deployed application, the applications' deployment information and any dependencies included therein, generate a new configuration information and communicate the new configuration information to a second application server that is used in deploying the application at the second application server;

        wherein said preprocessor performs the steps of:

        interrogating the deployed application at the first application server to find all ~~Java Naming and Directory Interface names~~ naming and directory interface entities present in the application;

        determining which of said ~~Java Naming and Directory Interface~~ naming and directory interface entities will be realized at runtime; and,

        parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another.

3.   (Cancelled)

4.   (Cancelled)

5.   (Currently Amended) A system to migrate a software application from one application server to a different application server, comprising:

        a first application server that includes an application deployed thereon;

        a preprocessor server component on the first application server used to interrogate the functionality of the deployed application, the applications' deployment information and any dependencies included therein, generate a new configuration information and communicate the new configuration information to a second application server that is used in deploying the

-3-

application at the second application server; and

~~The system of claim 1~~ wherein the application side defines any ~~Enterprise Java Beans~~ enterprise beans used in the application and the resources dependent thereon, and the server side defines management interfaces used by the application, data sources, and ~~JMS~~ messaging service queues.

6.   (Cancelled)

7.   (Currently Amended) ~~The method of claim 6 wherein said preprocessor performs the steps of:~~ A method for migrating a software application from one software application server to a different software application server, comprising the steps of:

        providing a first application server that includes a preprocessor and an application deployed thereon;

        using the preprocessor to interrogate the functionality of the application, the applications' deployment information and any dependencies included therein, generate new configuration information and to communicate the new configuration information to a second application server that is used in deploying the application at the second application server;

        wherein said preprocessor performs the steps of:

        interrogating the deployed application at the first application server to find all ~~Java Naming and Directory Interface names~~ naming and directory interface entities present in the application;

        determining which of said ~~Java Naming and Directory Interface~~ naming and directory interface entities will be realized at runtime; and,

        parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another.

8.   (Cancelled)

9.   (Cancelled)

10.   (Currently Amended) A method for migrating a software application from one software application server to a different software application server, comprising the steps of:

        providing a first application server that includes a preprocessor and an application deployed

-4-

thereon;

using the preprocessor to interrogate the functionality of the application, the applications'
deployment information and any dependencies included therein, generate new configuration
information and to communicate the new configuration information to a second application server
that is used in deploying the application at the second application server; and

~~The method of claim 6~~ wherein the application side defines any ~~Enterprise Java Beans~~
enterprise beans used in the application and the resources dependent thereon, and the server side
defines management interfaces used by the application, data sources, and ~~Java Messaging Service~~
messaging service queues.

11.   (Cancelled)

12.   (Currently Amended) ~~The computer readable medium of claim 11 wherein said preprocessor~~
~~performs the steps of:~~ A computer readable medium including instructions stored thereon which
when executed cause the computer to perform the steps of:

providing a first application server that includes an application deployed thereon and a
preprocessor;

interrogating the application's functionality, the applications' deployment information and any
dependencies included therein using the preprocessor, generating new configuration information,
and communicating the new configuration information to a second application server that is used in
deploying the application at the second application server;

wherein said preprocessor performs the steps of:

interrogating the deployed application at the first application server to find all ~~Java Naming~~
~~and Directory Interface names~~ naming and directory interface entities present in the application;

determining which of said ~~Java Naming and Directory Interface~~ naming and directory
interface entities will be realized at runtime;

parsing through both an application-side list, and a server-side list, and locating
dependencies that correlate with one another; and,

communicating the new configuration information that is used in deploying the application on
said second application server.

13.   (Cancelled)

-5-

14.   (Cancelled)

15.   (Currently Amended) <u>A computer readable medium including instructions stored thereon which when executed cause the computer to perform the steps of:</u>

<u>providing a first application server that includes an application deployed thereon and a preprocessor;</u>

<u>interrogating the application's functionality, the applications' deployment information and any dependencies included therein using the preprocessor, generating new configuration information, and communicating the new configuration information to a second application server that is used in deploying the application at the second application server; and</u>

~~The computer readable medium of claim 11~~ wherein the application side defines any ~~Enterprise Java Beans~~ <u>enterprise beans</u> used in the application and the resources dependent thereon, and the server side defines management interfaces used by the application, data sources, and ~~Java Messaging Service~~ <u>messaging service</u> queues.

16.   (Cancelled)

17.   (Currently Amended) ~~The system of claim 16 wherein said preprocessor performs the steps of:~~ <u>A system for readily deploying software applications from a first server to a second server, comprising:</u>

<u>a first server having an application deployed thereon;</u>

<u>a second server adapted to receive said application;</u>

<u>a preprocessor on said first server that interrogates the application's functionality, the application's deployment information as deployed on said first server, and any dependencies included therein, and generates or communicates a new configuration information, that is used in deploying the application at said second server;</u>

<u>wherein said preprocessor performs the steps of:</u>

interrogating the deployed application at the first application server to find all ~~Java Naming and Directory Interface names~~ <u>naming and directory interface entities</u> present in the application;

determining which of said ~~Java Naming and Directory Interface~~ <u>naming and directory interface</u> entities will be realized at runtime; and,

-6-

parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another.

18. (Cancelled)

19. (Cancelled)

20. (Currently Amended) <u>A system for readily deploying software applications from a first server to a second server, comprising:</u>

        <u>a first server having an application deployed thereon;</u>

        <u>a second server adapted to receive said application;</u>

        <u>a preprocessor on said first server that interrogates the application's functionality, the application's deployment information as deployed on said first server, and any dependencies included therein, and generates or communicates a new configuration information, that is used in deploying the application at said second server; and</u>

        ~~The system of claim 16~~ wherein the application side defines any ~~Enterprise Java Beans~~ <u>enterprise beans</u> used in the application and the resources dependent thereon, and the server side defines management interfaces used by the application, data sources, and ~~Java Messaging Service~~ <u>messaging service</u> queues.

21. (Cancelled)

22. (Currently Amended) ~~The method of claim 21 wherein said preprocessor performs the steps of:~~ <u>A method for readily deploying software applications from a first server to a second server, comprising the steps of:</u>

        <u>providing a first server having an application deployed thereon and a preprocessor;</u>

        <u>providing a second server adapted to receive said application;</u>

        <u>interrogating the application's functionality, the application's deployment information as deployed on said first server, and any dependencies included therein using the preprocessor, and generating or communicating a new configuration information, that is used in deploying the application at said second server;</u>

        <u>wherein said preprocessor performs the steps of:</u>

-7-

interrogating the deployed application at the first application server to find all ~~Java Naming and Directory Interface names~~ naming and directory interface entities present in the application;

determining which of said ~~Java Naming and Directory Interface~~ naming and directory interface entities will be realized at runtime;

parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another; and,

communicating said new configuration information that is used in deploying the application on said second application server.

23.   (Cancelled)

24.   (Cancelled)

25.   (Currently Amended) A method for readily deploying software applications from a first server to a second server, comprising the steps of:

providing a first server having an application deployed thereon and a preprocessor;

providing a second server adapted to receive said application;

interrogating the application's functionality, the application's deployment information as deployed on said first server, and any dependencies included therein using the preprocessor, and generating or communicating a new configuration information, that is used in deploying the application at said second server; and

~~The method of claim 21~~ wherein the application side defines any ~~Enterprise Java Beans~~ enterprise beans used in the application and the resources dependent thereon, and the server side defines management interfaces used by the application, data sources, and ~~Java Messaging Service~~ messaging service queues.

26.   (Cancelled)

27.   (Currently Amended) ~~The computer readable medium of claim 26 wherein said preprocessor performs the steps of:~~ A computer readable medium including instructions stored thereon which when executed cause the computer to perform the steps of:

providing a first server having an application deployed thereon and a preprocessor;

-8-

providing a second server adapted to receive said application;

interrogating the application's functionality, the application's deployment information as deployed on said first server, and any dependencies included therein using the preprocessor, and generating or communicating new configuration information, that is used in deploying the application at said second server;

wherein said preprocessor performs the steps of:

interrogating the deployed application at the first application server to find all ~~Java Naming and Directory Interface names~~ naming and directory interface entities present in the application;

determining which of said ~~Java Naming and Directory Interface~~ naming and directory interface entities will be realized at runtime;

parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another; and,

communicating the new configuration information that is used in deploying the application on said second application server.

28.   (Cancelled)

29.   (Cancelled)

30.   (Currently Amended) A computer readable medium including instructions stored thereon which when executed cause the computer to perform the steps of:

providing a first server having an application deployed thereon and a preprocessor;

providing a second server adapted to receive said application;

interrogating the application's functionality, the application's deployment information as deployed on said first server, and any dependencies included therein using the preprocessor, and generating or communicating new configuration information, that is used in deploying the application at said second server; and

~~The computer readable medium of claim 26~~ wherein the application side defines any ~~EJBs~~ enterprise beans used in the application and the resources dependent thereon, and the server side defines management interfaces used by the application, data sources, and ~~Java Messaging Service~~ messaging service queues.

-9-

31.   (Cancelled)

32.   (Currently Amended) A system to migrate a software application from one application server to a different application server, comprising:

a first application server that includes an application deployed thereon;

a preprocessor server component on the first application server used to interrogate the functionality of the deployed application, the applications' deployment information and any dependencies included therein, generate a new configuration information and communicate the new configuration information to a second application server that is used in deploying the application at the second application server; and

~~The system of claim 1 where in~~ wherein deployment information is sifted to see which application entities will be realized at runtime.

33.   (Cancelled)

34.   (Currently Amended) A method for migrating a software application from one software application server to a different software application server, comprising the steps of:

providing a first application server that includes a preprocessor and an application deployed thereon;

using the preprocessor to interrogate the functionality of the application, the applications' deployment information and any dependencies included therein, generate new configuration information and to communicate the new configuration information to a second application server that is used in deploying the application at the second application server; and

~~The method of claim 6~~ wherein the deployment information is sifted to see which application entities will be realized at runtime.

-10-